Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently amended) An arrangement for controlling communication between a client application executing on a client unit and an impermanently connected server in a network, comprising:

at least one client proxy at the client unit, the at least one client proxy operable, when the client unit is not connected for communication with the server, for receiving file transfer protocol (FTP) information from the client application executing on the client unit that requires a server response for the client application to continue operation, for storing the file transfer protocol (FTP) information, for generating a substitute response for the server response to allow the client application to continue operation, and for sending the substitute response to the client application; and

means for relaying the file transfer protocol (FTP) information from the at least one client proxy to the server responsive to a communications link between the client unit and the server being established, wherein the client proxy comprises:

means, responsive to receiving a file transfer protocol (FTP) file 'GET' command, for writing a local file at the client unit having a unique signature; and

means, responsive to the connectivity link between the client unit and the server being established, for obtaining a requested file from the server, for finding the local file at the client unit having the unique signature, and for over-writing the found local file with the obtained requested file.

2.   (Canceled)

3.   (Currently amended) The arrangement of claim 1, further comprising server proxy means coupled to the server for cooperating with the client proxy at the client unit to relay said file transfer protocol (FTP) information from the client unit to the server responsive to said connectivity link being established.

4.   (Currently amended) The arrangement of claim 3 further comprising messaging means coupled between the client unit and the server for relaying said file transfer protocol (FTP) information as messages across the connectivity link responsive to said connectivity link being established.

5-9.   (Canceled)

10.   (Currently amended) The arrangement of claim 1 further comprising means for notifying a user of the client unit of an outcome of the relaying of said file transfer protocol (FTP) information to the server.

11.   (Original) The arrangement of claim 1 wherein the client unit comprises a portable computing device.

12.   (Currently amended) A method for impermanent connectivity between a client unit and a server in a network, the method comprising:
        providing a client proxy at the client unit for receiving file transfer protocol (FTP) information to be sent to the server and for which a server response is to be received from the server, for storing said file transfer protocol (FTP) information, and for sending a substitute response for the server response; and
        relaying said file transfer protocol (FTP) information from the client proxy at the client unit to the server responsive to a connectivity link between the client unit and the server being established, wherein the steps of receiving, storing and relaying said information comprises:
                in response to receiving a file transfer protocol (FTP) file 'GET' command, writing a local file at the client unit having a unique signature; and

responsive to the connectivity link being established between the client unit and the server, obtaining a requested file from the server, finding the local file at the client unit having the unique signature, and over-writing the found local file with the obtained requested file.

13.   (Currently amended) The method of claim 12 further comprising providing a server proxy coupled to the server for cooperating with the client proxy to relay said file transfer protocol (FTP) information from the client unit to the server responsive to said connectivity link between the client unit and the server being established.

14.   (Currently amended) The method of claim 12 further comprising providing messaging means coupled between the client unit and the server for relaying said file transfer protocol (FTP) information as messages across the connectivity link responsive to said connectivity link being established.

15-19. (Canceled)

20.   (Currently amended) The method of claim 11 further comprising notifying a user of the client unit of an outcome of the relaying of said file transfer protocol (FTP) information to the server.

21.   (Original) The method of claim 11 wherein the client unit comprises a portable computing device.

22.   (Canceled)

23.   (Currently amended) A computer program product, comprising:

a computer readable data ~~carrier carrying~~ <u>storage medium storing</u> a computer readable program element for impermanent connectivity between a client unit and a server in a network, the computer program product comprising:

instructions for providing at the client unit, a client proxy for receiving <u>file transfer protocol (FTP)</u> information to be sent to the server and for which a server response is to be received from the server, for storing said <u>file transfer protocol (FTP)</u> information, and for sending a substitute response for the server response; and

instructions for relaying said <u>file transfer protocol (FTP)</u> information from the client unit to the server responsive to a connectivity link between the client unit and the server being established<u>, wherein the instructions for providing a client proxy comprises:</u>

<u>instructions, responsive to receiving a file transfer protocol (FTP) file 'GET' command, for writing a local file at the client unit having a unique signature; and</u>

<u>instructions, responsive to the connectivity link between the client unit and the server being established, for obtaining a requested file from the server, for finding the local file at the client unit having the unique signature, and for over-writing the found local file with the obtained requested file.</u>